Result of voting

Consultation Information:	
Consultation reference:	Conditionally accepted work items for Fortran 202Y
Consultation title:	Conditionally accepted work items for Fortran 202Y
Opening date:	2025-04-06
Closing date:	2025-05-03
Note:	At the 2024 meeting, three prospective work items were "conditionally accepted". INCITS/Fortran was requested to discuss these at a future meeting to provide direction to WG5 - this happened at the February 2025 meeting in Berkeley, California. Given the abbreviated (and end-of-week) schedule for the 2025 WG5 meeting, a request was made to provide INCITS/Fortran's guidance and to ballot WG5 members on formally accepting the work items so that development could start earlier. The three prospective work items are as follows: US20 - Add SCAN and CO_SCAN intrinsic procedures. (The names would likely change.) See j3-fortran.org/doc/year/23/23-235r2.txt for more details. The INCITS/Fortran vote on this was 11 in favor, 0 opposed. Comments made at the time: These are difficult for users to implement on their own with acceptable performance. C++ already has these, and some vendors have already implemented them for their own use. US22/DIN02 - Unions in interoperable types. See j3-fortran.org/doc/year/24/24-117.txt It was pointed out that trying to emulate this feature with independently declared types and TRANSFER loses the feature of automatically sizing the union to the maximum of the overlaid types. Most compilers already support unions using a

	nonstandard syntax. The INCITS/Fortran vote on this was 7 in favor, 0 opposed, 4 undecided. DIN01 - Collectives over a specified team. See N2230.pdf The INCITS/Fortran vote on this was 10 in favor, 0 opposed, 1 absent

Experts responses:	
Votes cast (26)	Bonachea, Dan Brady, Chris Mr Chen, Daniel Mr Chivers, Ian Mr Clune, Thomas Dr Cohen, Malcolm Mr Cook, Brandon Dr Cornille, Brian Mr Hammond, Jeff Dr Haveraaen, Magne Ms Iwashita, Hidetoshi Mr Johnson, Ted Mr Klimowicz, Gary Mr Leair, Mark Mr Lionel, Steve Mr Ma, Chengqing Mrs Maddhipatla, Raghu Mr Menard, Lorri Ms Muxworthy, David T Mr Rasmussen, Katherine Rasmussen, Soren Dr Reid, John K Dr Rouson, Damian Dr Steidel, Jon Mr Takata, Masayuki Mr Thomas, Paul Richard Mr
Votes not cast (58)	Adelstein Lelbach, Bryce Mr Agbere, Dabiyyah Ahn, So Young Ms Ash, Bill Mr Barra, Lynn Ms Bernholdt, David Mr Brito Gadeschi, Gonzalo Mr Budiardja, Reuben Dr Bullock, Austin Mr Certik, Ondrej Mr Chandramohan, Kiran Mr Costa, Timothy Dr Curcic, Milan Dr Elsesser, Gary

Georgiadou, Antigoni
Gibney, Jonathon
Gorelik, Alla Dr
Green, Ron
Hemstad, Jake Mr
Henning, Paul Dr
Hoemmen, Mark Dr
Huhn, William Mr
Jin, Henry Mr
Jung, Bo Mr
Jung, HaeSun Prof
Kadhem, Hugh
Kamatsuka, Shun Mr
Kamenoue, Tsuyoshi Mr
Kwon, Seung Uk Mr
Larkin, Jeff Mr
Lee, Yong Woo
Liber, Nevin Mr
Long, Bill Dr
Mangudi, Divya Mr
Markus, Arjen Dr
Myeong, Bongsik Mr
Nicely, Matthew Mr
Niebler, Eric Mr
Okano, Shinichi
Otken, John
Pettey, Lucas Dr
Rasmussen, Craig Dr
Saldivar, Miguel
Sato, Fumitoshi Mr
Sato, Hiroyuki Dr
Schweitz, Eric
Shafran, Aury
Sharp, Philip Dr
Shterenlikht, Anton Dr
Sircombe, Nathan John Dr
Sleightholme, Jane Ms
Sottile, Matthew Dr
Spittle, Deborah J. Ms
Sukhomlin, Vladimir Mr.
Szeremi, Vendel
Tschirhart, Zachary Mr
Wei, Weile Dr
Zumwalde, Melanie

Questions:	
Q.1	"Should US20 (SCAN/CO_SCAN) be approved for the Fortran 202Y work item list?"

Q.2	"Should US22/DIN02 (Unions in interoperable types) be approved for the Fortran 202Y work item list?"
Q.3	"Should DIN01 (Collectives over a specified team) be approved for the Fortran 202Y work item list?"

Answers to Q.1: "Should US20 (SCAN/CO_SCAN) be approved for the Fortran 202Y work item list?"		
20 x	**Yes**	**Bonachea, Dan** **Brady, Chris Mr** **Chen, Daniel Mr** **Chivers, Ian Mr** **Clune, Thomas Dr** **Cook, Brandon Dr** **Cornille, Brian Mr** **Hammond, Jeff Dr** **Iwashita, Hidetoshi Mr** **Johnson, Ted Mr** **Klimowicz, Gary Mr** **Leair, Mark Mr** **Lionel, Steve Mr** **Ma, Chengqing Mrs** **Maddhipatla, Raghu Mr** **Rasmussen, Katherine** **Rasmussen, Soren Dr** **Rouson, Damian Dr** **Steidel, Jon Mr** **Takata, Masayuki Mr**
5 x	**No**	**Cohen, Malcolm Mr** **Menard, Lorri Ms** **Muxworthy, David T Mr** **Reid, John K Dr** **Thomas, Paul Richard Mr**
1 x	**Undecided**	**Haveraaen, Magne Ms**

Answers to Q.2: "Should US22/DIN02 (Unions in interoperable types) be approved for the Fortran 202Y work item list?"		
20 x	**Yes**	**Bonachea, Dan** **Chen, Daniel Mr** **Chivers, Ian Mr** **Clune, Thomas Dr** **Cook, Brandon Dr** **Cornille, Brian Mr** **Hammond, Jeff Dr** **Iwashita, Hidetoshi Mr** **Johnson, Ted Mr** **Klimowicz, Gary Mr** **Leair, Mark Mr**

		Lionel, Steve Mr Ma, Chengqing Mrs Maddhipatla, Raghu Mr Menard, Lorri Ms Rasmussen, Soren Dr Rouson, Damian Dr Steidel, Jon Mr Takata, Masayuki Mr Thomas, Paul Richard Mr
5 x	No	Brady, Chris Mr Cohen, Malcolm Mr Haveraaen, Magne Ms Muxworthy, David T Mr Reid, John K Dr
1 x	Undecided	Rasmussen, Katherine

Answers to Q.3: "Should DIN01 (Collectives over a specified team) be approved for the Fortran 202Y work item list?"		
23 x	Yes	Bonachea, Dan Brady, Chris Mr Chen, Daniel Mr Chivers, Ian Mr Clune, Thomas Dr Cohen, Malcolm Mr Cook, Brandon Dr Cornille, Brian Mr Hammond, Jeff Dr Haveraaen, Magne Ms Iwashita, Hidetoshi Mr Johnson, Ted Mr Klimowicz, Gary Mr Leair, Mark Mr Lionel, Steve Mr Ma, Chengqing Mrs Maddhipatla, Raghu Mr Menard, Lorri Ms Rasmussen, Katherine Rasmussen, Soren Dr Rouson, Damian Dr Steidel, Jon Mr Takata, Masayuki Mr
3 x	No	Muxworthy, David T Mr Reid, John K Dr Thomas, Paul Richard Mr
0 x	Undecided	

Comments from Voters

Expert:	Comment:	Date:
Brady, Chris Mr	***Comment***	2025-05-02 11:57:17
Not a high priority and potentially risky		
Cohen, Malcolm Mr	***Comment***	2025-04-25 01:27:20
CO_SCAN in particular is not part of C++; its likely benefit is too small to outweigh even a small cost. SCAN was part of HPF, and several vendors implemented it, and the user appreciation was so underwhelming that they have all dropped it since. These are inherently unsafe. Fortran does not need more unreliability.		
Haveraaen, Magne Ms	***Comment***	2025-04-08 16:39:32
Rather than as intrinsics, scan/co_scan might better be supported as part of a standard template-generic library for Fortran. The interoperability of Fortran intrinsics require a much more detailed semantics of the bit-patterns of said types than given in the Fortran standard. Specifcially: KIND arguments to intrinsics are currently not standardised across processors. Using union types without such standardisation will yield non-portable code, going against the purpose of standardising a union type.		
Menard, Lorri Ms	***Comment***	2025-04-24 21:05:33
While this might map directly onto an (obsolete) HPF construct, there hasn't been a compelling use case yet.		
Muxworthy, David T Mr	***Comment***	2025-05-01 18:13:25
It is more important to adhere to the schedule in N2232 than to add an item of debatable widespread utility. The feature is reported to be unsafe. It is important to adhere to the schedule in N2232. A review of all the items in N2234 would be desirable.		
Rasmussen, Katherine	***Comment***	2025-04-07 22:34:06
I find some arguments for this work item to be persuasive, as well as some arguments against this work item. Hence, I am undecided at this time.		
Reid, John K Dr	***Comment***	2025-05-02 11:45:58
Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list. Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list. Insufficient progress has been made on the present list of work items. This item could be added after the removal of a large item from the present list.		

Takata, Masayuki Mr	***Comment***	2025-05-02 06:34:31
I hate to extend use of equivalence association, but take this as a necessary evil.		
Thomas, Paul Richard Mr	***Comment***	2025-04-30 08:27:07
There are too many items on the existing work list. That said, SCAN/CO_SCAN could be easily implemented. Vendors have already implemented this feature for DEC compatability. Same comment as for SSCA/CO_SCAN although I am less sure of the ease of implementation.		